NEW JERSEY MEADOWLANDS COMMISSION

1 DEKORTE PARK PLAZA
LYNDHURST, NJ 07071
WWW.NJMEADOWLANDS.GOV

NJMC/MERI GIS

LAND PARCEL DATA MODEL

Prepared by: Alex Marti, MERI-GIS
January , 2005

NJMC / MERI GIS

LAND PARCEL DATA MODEL

1. INTRODUCTION

On September 30, 2004 the Land Parcel Data Model was successfully implemented into the Spatial Database Engine (SDE) after being tested in Personal Geodatabase. This manual outlines the implementation of this Model in ArcSDE at the MERI-GIS Department as a part of the Municipal Outreach Program (MAP). The goal of the ArcSDE project is the successful migration of existing Shapefiles (SHP) and ArcInfo Coverages (COV) into Geodatabase (spatial database) that utilizes ArcSDE as the gateway between the GIS and the relational database management system (RDBMS). The conversion of GIS data into a Geodatabase will eliminate the duplication of data and limit unnecessary updates by using the Geodatabase schema. One important advantage from the conversion is taking advantage of the Geodatabase functionality, including the ability to model spatial features and their behavior with subtypes, domains, topology rules and relationships.
Some of the other advantages that a Geodatabase has over conventional GIS formats are:

- Centralized spatial and attribute storage
- Multiple user-editing sessions
- Multiple customization options
- Easy to use/standard behavior rules including domains, subtypes, topology and networks
- Easy to import, export and share schema with or without data
- Provide a faster and consistent access to the data and reliable storage to the server.
- Enforce data-integrity rules and control access through security

The MERI-GIS Land Parcel Data Model is based from the ESRI Land Parcel Model. The current database is roughly 205Mb and contains the Parcel , Block, Land Use, Zoning, Encumbrance and Buildings geo-spatial data layers for the MAP Program-Phase I Towns (Secaucus, Carlstadt, Moonachie and Teterboro). The parcel data-layer contains about 6700 parcels, and are maintained in the Geodatabase. As more towns are completed, this number will significantly increase.
The Municipal Assistant Program (MAP) Dataset includes subtypes, domains, tables and relationships that model spatial features and their behavior. The domains, tables and their relationships are related to the current parcel data-layer. They are described in the appendix I and II.

Note that the current domains will change since new municipalities will be added. For other data that populate the Enterprise GIS server refer to the Migration Plan Report (inventory).

2. LAND PARCEL DATA MODEL IMPLEMENTATION

Details of the Data development and the Land Parcel Data Model implementation are described in the following sections:

1. Municipal Data sets development/update.

Data development for Parcel, Block, Land Use, Zoning, Encumbrance and Buildings are maintained for each Municipality. Each geo-spatial data layer developed must be accurately georeferenced using adjacent towns and other existing data. You can use topology rules to determine potential inconsistencies with the data, and, then fix potential errors in ArcMap. A topology is a set of relationships that defines how the features in one or more feature classes (FC) share geometry. Be sure to assign a proper rank for every FC; that will determine how much each FC will move once the topology is validated.

Notice that you **can** create more than one topology set per Feature Data Set but you **cannot** use the same feature classes that are already participating in another topology set. Once you are satisfied with your Topology you can validate it. By validating topology, will check if features violate any rules that were defined in the topology. After corrections, it is recommended to validate again.

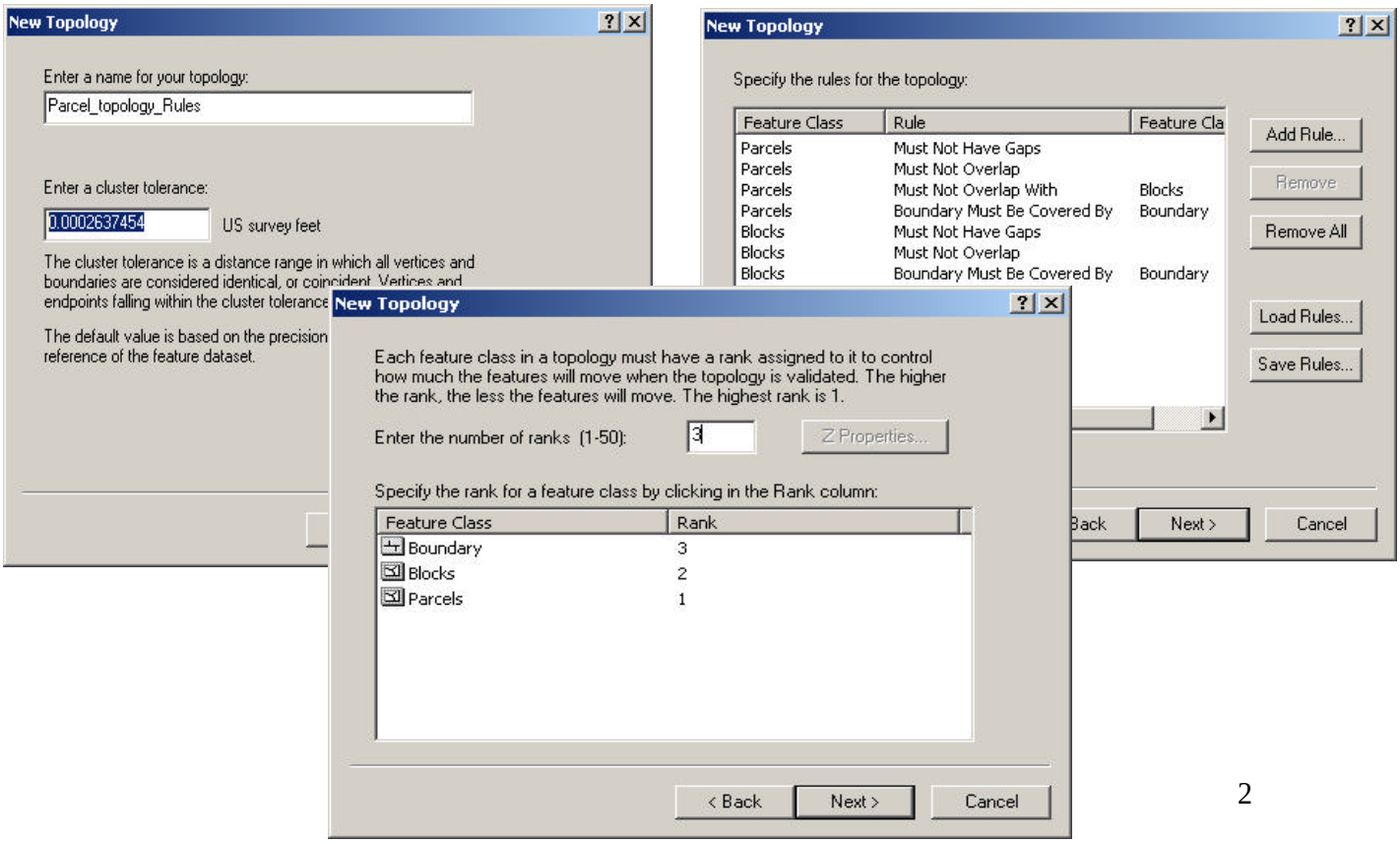


Figure 1. Example of Topology Rules for parcel, block and Boundary Feature class

2. MOD4 property Tax List 2004 Join.

When Joining the MOD4 tax information each record in the *Join Info Table* is matched to each record of the *Input Info Table* when the *Relate Item* values are equal. The item values from the two records are copied to the *Intput Table*. The resulting *Output Table* will contain the information of both the parcel and the MOD4 databases. In our case, we are going to join the input MOD4 data into our Parcel table, which is the Output Table. The Project Key will be the relate item to perform the join. Note that some data will not match due to the inconsistency of the MOD4 database. A cleaning process of the relate item will be needed to maximize the percentage of the records matching. Since the Input Info Table is a text file we will use the Prop-I-Grator application to convert the database into a DBF file and create the project key that will be used as the relate item with the parcel SHP.

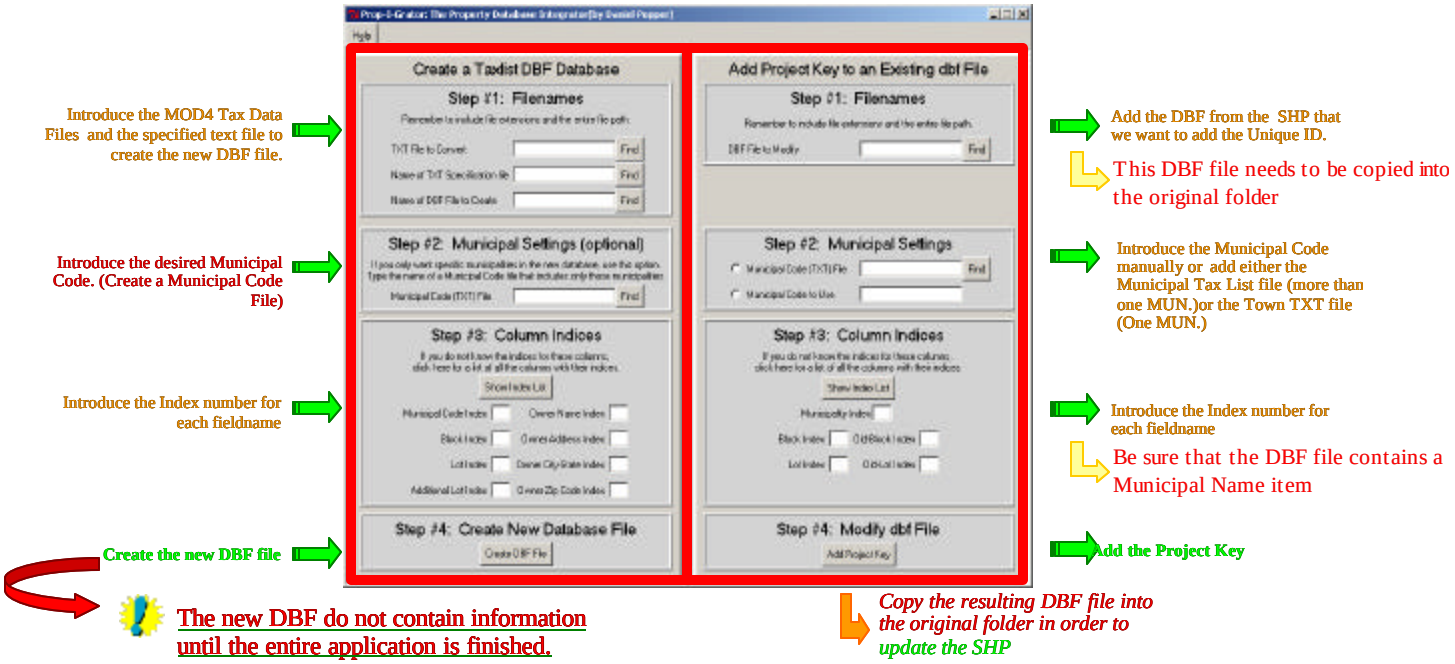


Figure 2. Example of the process for a DBF creation and Project Key addition

Note: ***Refer to the Property Database Integrator Manual for more detail***

3. Personal Geodatabase Implementation / Set up

In order to provide a better quality product we implemented a Personal Geodatabase in ArcCatalog for testing purposes. Personal Geodatabases are user defined relational databases that contain different types of geographic information such as feature classes or tables. Feature classes can be organized into feature datasets. Feature classes store geographic features that are represented as points, lines, polygons and their attributes. All feature classes in a feature dataset share the same coordinate system. There is a designated Personal Geodatabase located in the H:\ drive for all the MAP FC. The Copy of District Attempt 5 MDB (temporary Name) contains all the FC included as a part the Phase I towns under the MAP project. This Geodatabase is composed of 3 Feature Datasets (Land Records, Environmental and Encumbrance) where the FC are stored depending on the type of data that they contain, as well as 2 Tables (Owner Table and Intermediate Table). Many objects in a Geodatabase can be related to each other. In this Personal Geodatabase, the owner and the parcel information are related through a bulk table (intermediate table). To explicitly define the relationships between objects in a Geodatabase, you must create Relationship classes (see section 5, Building Relationships). The relationship classes for the MERI-GIS Land Parcel Model are detailed in the Appendix II.

The implementation of the Personal Geodatabase require an accurate order of steps described as follows:

- Once you have created and defined your Personal Geodatabase and Feature Dataset you need to upload the desired FC's. For those FC's that will work as a master FC, you need to create an empty FC and upload each coverage one by one. Make sure that the extent of the Feature Dataset is big enough to fit all the data that you will upload. Be far-sighted. In order to be consistent, you can use the predefined extend located in H:\TownEx.shp
- It is recommended to create a schema for your master FC before uploading any data. Remember that both type and length of each attribute from uploading data need to match with the attributes defined by the user in the master FC.
- The Parcel FC needs to have a ParcelID and an OwnerID as static fields in order to be uploaded. Both fields will be used as a primary and foreign key respectively in the relationships later on. **Do NOT** change the ObjectID for this purpose.
- Make sure that the data that you want to upload does not contain any raster field such as a photo field. That will produce performance errors in future processes like export into SHP or COV.
- Once all the FC's are located in the right Feature Dataset you need to create Domains and Subtypes. A **Domain** is a declaration of acceptable attribute values**.** Domain management

involves the following steps: Create the domain, add values to the domain or set the range of values for the domain and associate the domain with a table or an attribute of a feature class. When an attribute domain is associated with a table or feature class, an attribute validation rule is also created in the database. This attribute validation rule describes and constrains the valid values of a field type. One attribute domain can be associated with multiple fields in the same table, feature class, or subtypes as well as in multiple tables and feature classes. (See Appendix I for details on the NJMC Land Parcel Model Domains). A **Subtype** is a way to group features of one feature class into subsets based on an attribute value (See fig. 3).

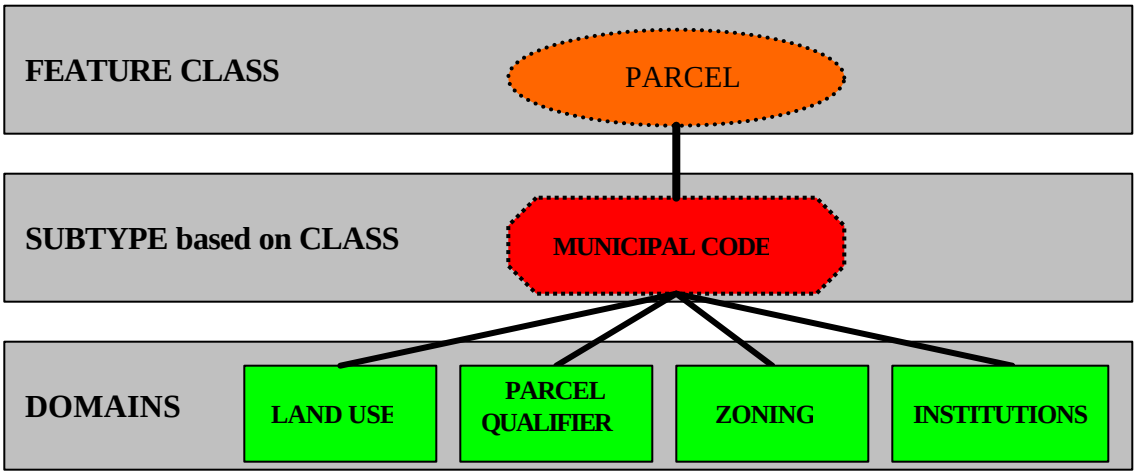


Figure *3. Domains and Subtypes for Parcel*

4. Tables. Design and maintenance.

Once the MOD4 data is joined to the parcel database, we need to extract the owner information and perform a well-maintained owner database that will be used for the 14 municipalities under the MAP. For this purpose, we need to extract the owner information by exporting the parcel database into a table by using ArcMap. This way, we will have a database with all the necessary owner information and with the pre-defined ParcelID created in the parcel FC.

With the owner database table separated from the parcel FC, we can delete the extra attributes that we will not need. Once that is done, we can start cleaning the owner information by correcting duplicate misspelled owners or by making sure that the owner address, city-state and zip code are consistent for those properties owned by the same owner. At the end of the cleaning, we should have multiple owners, sometimes repeated, but one consistent entry (same owner info: name, address, City-State, Zip Code) for each of them.

Next, we will need to assign an **OwnerID**. Make sure that for those repeated entries (owners), you

have assigned the same OwnerID. Once that is done, we can join the database back to the parcel FC in order to get the OwnerID in the Parcel FC.

Without deleting any field, we have to export the Parcel FC (joined) into a database table to obtain the **Intermediate Table** (also called bulk table). The intermediate table will be contained, in this fashion, with the same number of entries that the Parcel FC has, and thus, making a 100% match in the Parcel-to-Intermediates Relationship. The fields that should be present in this table are: *ParcelID, OwnerID and Project Key*. Any change done in any of these attributes must be reflected in both the parcel FC and intermediate table.

There is still one more step to do before finalizing the Owner Table. This table needs to be **summarized** by the OwnerID in order to get a unique owner database, whereby each owner is listed one time and assigned a unique OwnerID.

Once all this is completed, we can append the owner information for other towns (Phase II, etc…). It is very important to look for duplicate owners in order to maintain a unique entry for each owner. Therefore, we need to compare, once the MOD4 data that we want to upload is clean, to see if the owner we are going to upload has already an entry.

For further information about this section see manual: Owner Project (October, 2004)

Note: In order to edit any of these tables, remember to **Register with Geodatabase,``** "the table that you need to update."

5. Building Relationships.

Relationships are one of the key components of the Land Parcel Data Model. They let you use attributes stored in a related object to symbolize, label, or query a feature class and, above all, let you relate data from one table to another, from one FC to a table or, from two FC's as long as they share common fields. This will help us to optimize the updating process for our datasets. Each relationship can be built in many different ways depending on the needs of the user. In this way we can find:

1. **Simple peer-to-peer or Composite relationships**. While *a simple peer-to-peer* relationship is used when two objects in the database can exist independent of each other (ex: a parcel is deleted in the Parcel FC –origin- but the related information in the intermediate table –destination- stays); the *composite* relates the object lifetime (when deleting a parcel in the owner table –origin- the related information in the intermediate table –destination- is deleted too).
2. **Cardinality of the Relationship**. The cardinality of the relationship is defined depending on how many records are related between the origin and destination table/feature class. In a *one to one* cardinality, each feature in the origin table can be related to zero or one features in the

destination table. In a *one to many* cardinality, each feature in the origin table can be related to one or several features in the destination table. Finally, in a *many to many* cardinality, several features in the origin table can be related to one or several features in the destination table.

While a simple peer-to-peer relationship is by definition a *one-to-one* relationship, composite relationships always have a *one-to-many* cardinality. When you create a one-to-many relationship, whether simple or composite, the "one" side of the relationship must be the origin and the "many" side must be the destination.

3. **Direction of the Relationship**. The messages can be propagated by four different ways between the related objects. While the *forward* path label describes the relationship when navigated from the origin to the destination, the *backward* path label describes the same relationship when navigating from the destination to the origin. We use *both* when we want the messages to be propagated from both origin to the destination and destination to the origin. When selecting *none*, no messages are propagated.
4. **Selecting the Primary Key and the foreign key**. Generally the primary key is the object identifier field. If we are dealing with a 1-to-Many relationship, you will also need to select the foreign key in the destination table/feature class. The identifier fields will be those that we will use as the relate fields in the relationship.

In order to implement the NJMC Land Parcel Data Model we will require two different relationships:

- The **Parcel to Intermediate Relationship (R1)**, which relates the parcel FC located in the land records dataset, and the Intermediate table located under the MERILAB spatial database engine (Sde). The R1 is defined as a *Composite*; *1-to-Many* cardinality and *both* directions relationship using the ParcelID attribute field from the Parcel FC and the ParcelID attribute field from the intermediate table.
- And the **Owner to Intermediate Relationship (R2)**, which relates the owner and the intermediate tables located under the MERILAB.sde. The R2 is defined as a *Simple peer-to-peer*, *1-to-Many* cardinality and *Forward* direction relationship using the OwnerID attribute field from the owner table and the OwnerID attribute field from the intermediate table.

Note: In order to create the desired relationship some criteria must be followed. We must be sure that the specs of field that we are trying to use as a primary and foreign key match the type of field, width and precision. If those criteria are not followed, the desired field will not be listed as an option in the last step. We need to consider as well that the location of the relationships is limited if we are using more than one field from a table/feature class in different relationships. In this way, since the ParcelID and the OwnerID field from the Intermediate Table are being used, the Parcel To Intermediate Relationship will be located under the MAP Feature Dataset and the Owner To Intermediate Relationship will be located directly under the MERILAB.sde.

In order to grasp these steps, refer to the following figure for a detailed explanation.

1.

R1: Parcel To Intermediate **R2: Owner To Intermediate**

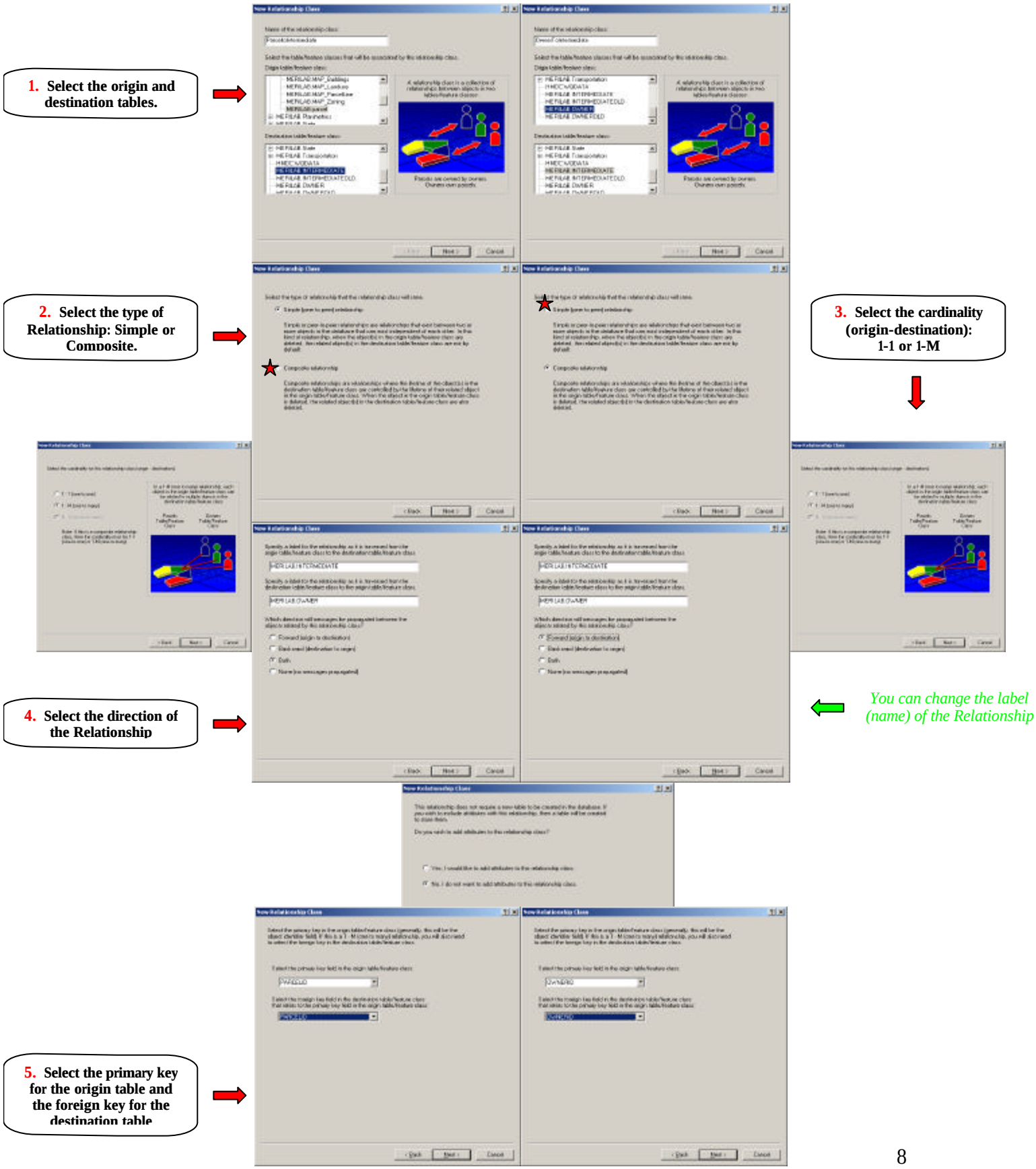


Figure 4. Relationships for the Land Parcel Data Model

SKETCH OF THE PROCESS

1. TOWN PARCEL FEATURE CLASSES AND OTHER DATA SETS.
    a. Parcel
    b. Block
    c. Land Use
    d. Zoning
2. JOIN MOD4 INFO
    a. Use ProjectKey
3. MATCH LINE WORK BETWEEN TOWN FC (GEOMETRY)
    a. Use topology
4. CREATE A FEATURE DATASET IN PERSONAL GEODATABASE AND DEFINE COORDINATE SYSTEM
    a. Create boundaries of the coordinate system for the entire 14 municipalities (use H:\TownEx.shp)
5. CREATE SCHEMA OF THE MASTER FC's INTO A PERSONAL GDB
6. CREATE FEATURE CLASSES IN THE FEATURE DATASET
7. CLEAN FIELDS AND MAKE THEY MATCH WITH THE SCHEMA
    a. DELETE THE PHOTO RASTER FIELD FROM THE SCHEMA. THE PARCEL. SDE DOES NOT TAKE RASTER FIELDS.
    b. CREATE PARCELID FIELD
8. UPLOAD FC's INTO MASTER FC/CREATE STATIC PARCEL ID
9. CREATE DOMAINS
10. CONNECT FIELDS TO THE DOMAIN
11. EXPORT ATTRIBUTE TABLE FROM PARCEL
12. KEEP OWNER INFORMATION, PROJECT KEY AND PARCELID AS A STATIC FIELD
13. CLEAN DATA AND DUPLICATES EDITS
14. ASSIGN AN UNIQUE OWNER ID PER OWNER (DO NOT SUMMARIZE YET)
15. JOIN BY PARCELID TO GET THE OWNER ID INTO THE PARCEL- Delete extra fields
16. EXPORT OWNER TABLE TO CREATE INTERMEDIATE TABLE
17. KEEP ONLY PARCELID, OWNERID AND PROJECTKEY
18. SUMMARIZE THE OWNER TABLE TO GET FINAL OWNER TABLE
19. REGISTER THE TABLES WITH GEODATABASE
20. CREATE RELATIONSHIP 1 (R1) BETWEEN PARCEL AND INTERMEDIATE

TABLE AS A COMPOSITE; BOTH; 1-M BY PARCELID-PARCELID

21. CREATE RELATIONSHIP 2 (R2) BETWEEN INTERMEDIATE AND OWNER TABLE AS A SIMPLE, FORWARD; 1-M; OWNERID-OWNERID
22. UPLOAD FC AND TABLES TO SDE
23. RECREATE RELATIONSHIPS IN THE SAME WAY CREATED IN A PERSONAL GEODATABASE.
24. CREATE VERSIONS, RECONCILE AND POST

3. SDE GEODATABASE IMPLEMENTATION

Once you have built and test your parcel model in Personal Geodatabase it is time to upload all the data into SDE. Even most of the steps for the creation of the model are similar in SDE; there are some important differences that we need to consider when building the model in SDE.

2. Versions.

Versioning lets users simultaneously create multiple, persistent representations of the database without data replication. Users can edit the same features or rows without explicitly applying locks to prohibit other users from modifying the same data. If, by chance, the same features are modified, a conflict resolution dialog box guides the user through the process of determining the feature's correct representation and attributes.

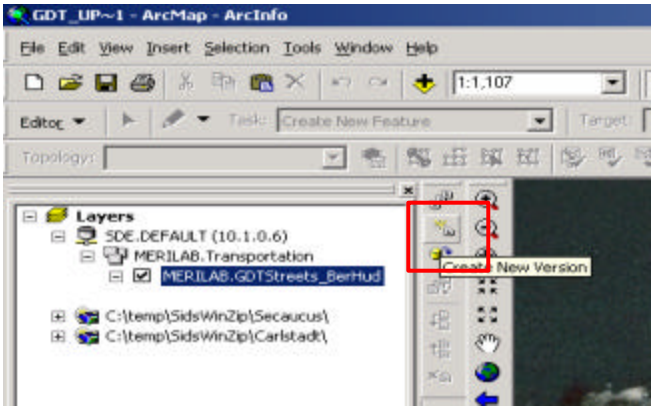
Initially, the database consists of one version named DEFAULT that is owned by ArcSDE administrator user. The new versions that are created are always based on an existing version. When the new version is created, it is identical to the version from which it was derived. Over time, the versions will be different as changes are made to the parent version and the new version.

1.1 Create a version

We can create versions from ArcCatalog and ArcMap. Both are valid. We will create our versions from ArcMap. **Note** that from now on, all edits that will be attempted on SDE will be final. There is no UNDO bottom, so please be very careful when you do this.

In order to create a version using ArcMap follow the follow steps:

1. Add the Versioning Toolbar to the map.
2. Click the Create New Version button. At least one version is required to be in ArcMap prior to the command becoming enabled.

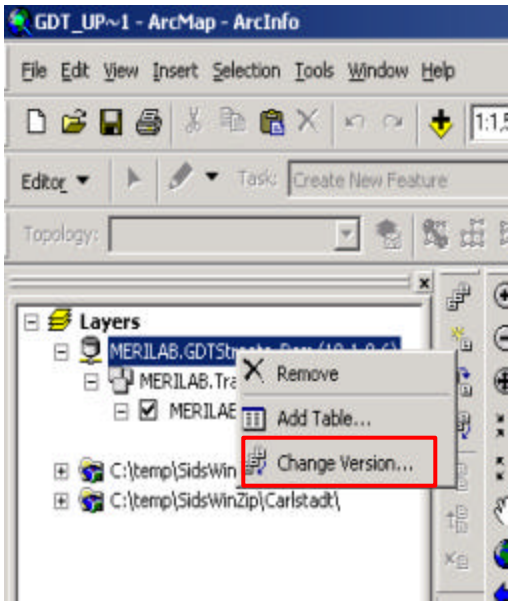


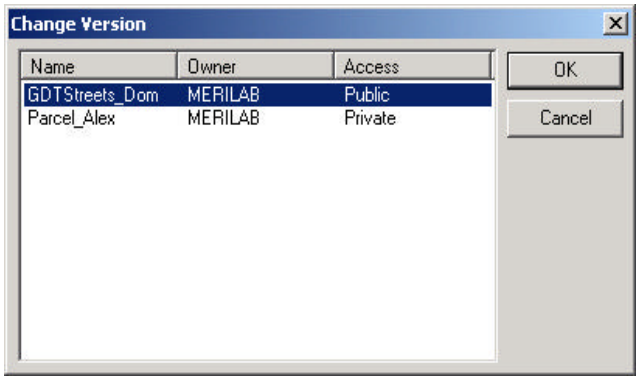
3. A second window will pop-up. Click the Parent Version dropdown arrow and click the parent version from which you want to create the new version. In our case we are dealing with **SDE.default**
4. Type the new version name. When naming, be sure to use the following naming convention: *NameofFC_UserName (ex: GDTStreets_Dom)*
5. Optionally, type a description.
6. Click the appropriate permission type.
7. Click OK.

Note that you can use the version description to provide additional information regarding the version's purpose. In the Version Manager dialog box, you can sort versions by clicking a column heading.

Optionally, if you aren't currently editing, check the check box to switch the parent version to the new version.

After creating the version, and if a user did not check the switch to new version, the user needs to do so. Therefore, right click on the database in the table of contents and change the parent version to the new version by clicking on *change version* and selecting the new one. This step is very important because we **do NOT** want to modify the parent version, which could have an effect with user's versions.



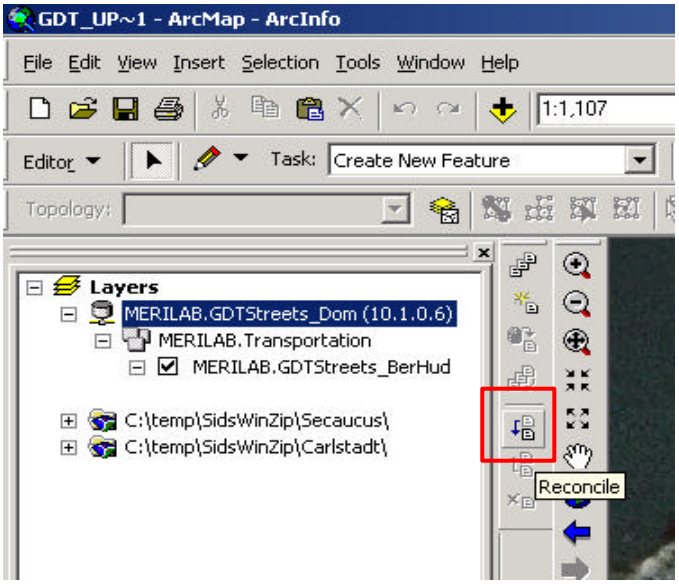


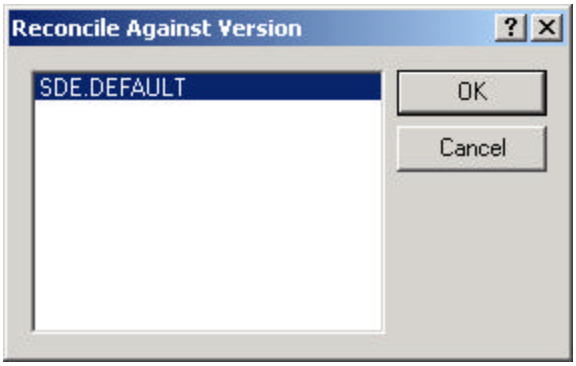
1.2 Delete a version

When **deleting versions** in ArcMap we have to consider that only the version's owner can rename, delete, or alter the version. Also, a parent version can't be deleted until all dependent child versions are deleted. Versions are not affected by changes occurring in other versions of the database.
In order to delete a version we just have to open the Version manager, right click on the version that we want to delete and select delete version. Before deleting a version make sure first that you have reconciled and posted the changes into the parent version. If not, you will loose any changed performed (which differs from the parent version).

1.3 Reconcile and Post

Reconcile merges all modifications between the current edit version and a target version (default). Any differences between the features in the target version and the features in the edit version are applied to the edit version. Differences can consist of newly inserted, deleted, or updated features. The reconcile process detects these differences and discovers any conflicts. Reconciling happens before posting a version to the parent version..

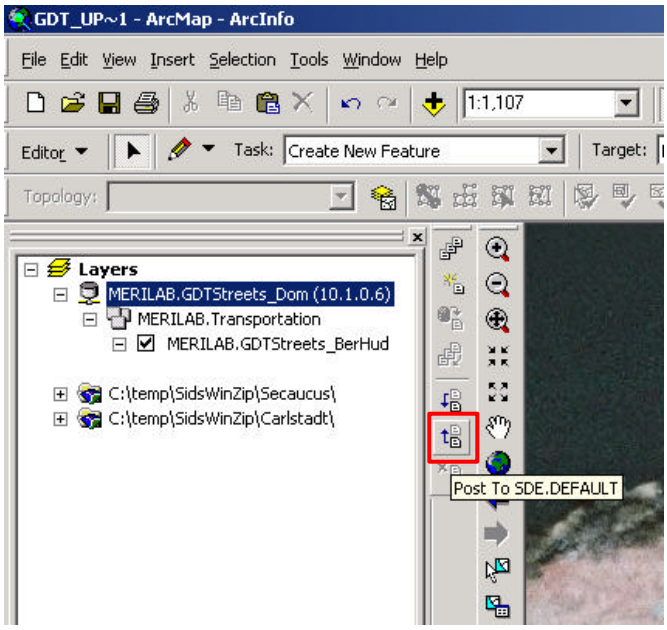




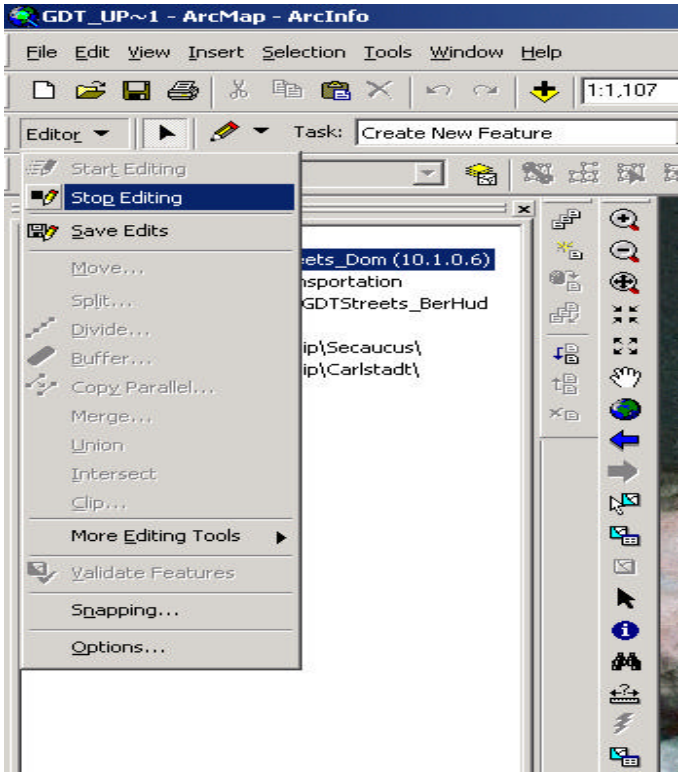
.

Posting synchronizes the edit version with the reconciled version and saves the data.

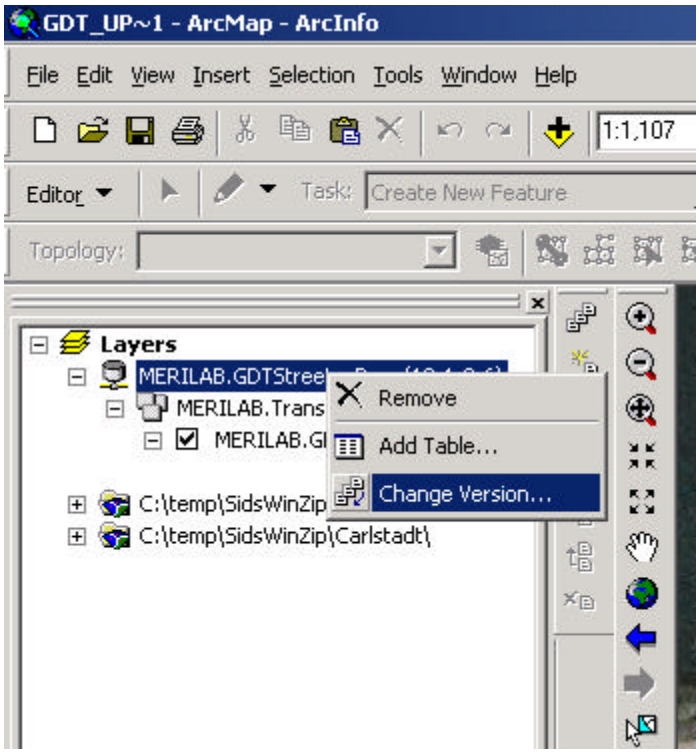
You have to notice that posting can't be undone, as you are applying changes to a version that you are not currently editing. If the reconciled version is modified between reconciling and posting, you will be notified to reconcile again before posting.



After posting, make sure that you stop editing



To make sure if your edit are successful, change your version to the SDE.DEFAULT. Click on several features that you have edited, and double-check your edits.
Lastly, once everything is complete and are 100% accurate, you can notify your SDE admin to compress the geodatabase (Merilab.sde). In order to do this, you must login as SDE administrator. Also, make sure ALL versions in the geodatabase (SDE) are unregistered before doing this.



APPENDIX I. MERI-GIS PARCEL DATA MODEL DOMAINS.

DOMAINS	Type	Code	Description
Institutions	Restaurants	RES	Institutions and Facility Descriptions
	Hospitals	HOS	
	Schools	SCH	
	Hotels & Motels	H&M	
	Gas Stations	GST	
	Fire Department	FDP	
	Banks	BNK	
	Church	CHU	
	Library	LIB	
	Parks	PRK	
Parcel Qualifier	Meadowlands District	MD	Qualifier Description
	Outside Meadowlands District	OMD	
	Meadowlands District & Outside	MD-OMD	
Land Use Categories	Altered Lands	AL	Land Use Descriptions and Layers
	Commercial Office	CO	
	Commercial Retail	CR	
	Communication Utility	CU	
	Hotels and Hotels	H&M	
	Industrial	IND	
	Industrial Commercial Complex	ICC	
	Multiple Uses	MU	
	Public/Quasi Public Service	PQP	
	Recreational Land	RL	
	Residential	RES	
	Transportation	TRS	
	Unclassified	000	
	Vacant Land	VC	
	Water	WAT	
	Wetlands	WET	
Zoning Categories	Aviation Facilities	AV	Zoning Description and Labels
	Commercial Park	CP	
	Environmental Conservation	EC	
	Heavy Industrial	HI	

	Highway Commercial	HC	
	Intermodal A	IA	
	Intermodal B	IB	
	Light Industrial A	LIA	
	Light Industrial B	LIB	
	Low Density Residential	LDR	
	Neighborhood Commercial	NC	
	Multiple Uses	MU	
	Parks and Recreation	P&R	
	Planned Residential	PR	
	Public Utilities	PU	
	Redevelopment Area	RA	
	Regional Commercial	RC	
	Roads, Rails, ROWs	RRR	
	Sports and Exposition	SEA	
	Transportation Center	TC	
	Unclassified	000	
	Waterfront Recreation	WR	
	Water	WAT	
	Commercial Zone C	CZC-SECA	
	Light Industrial Zone 1	LI1-SECA	
	Residential Zone A	RZA-SECA	
	Light Industrial & Distribution Zone	LID-TET	
	Zone Redevelopment Area 1	RA1-TET	
	Zone Redevelopment Area 2	RA2-TET	
Parcel Type	Private	PRV	Parcel Type Description
	Public	PUB	
	Public / Quasi Public	PQP	
	Unclassified	000	
Encumbrance Type	Planting	PLT	Encumbrance Type Description
	Rail Road	RXR	
	Road	RD	
	Utility	UTIL	
	Water	WAT	

	Unclassified	000	
Encumbrance Type	Deed	DEE	Encumbrance Area Type Description
	Lease	LEA	
	Prescriptive	PRE	
	Right of Way	ROW	
	Unclassified	000	
Boundary Type	County	CO	Boundary Type Description
	Municipal	MUN	
	State	NJ	
	Meadowlands District	NJMC	
Line Type	Others	OTH	Line Type Description
	Unclassified	000	

Table 1. Current Domains for MERI-GIS Parcel Data Model

APPENDIX II. MERI-GIS LAND PARCEL DATA MODEL RELATIONSHIPS.

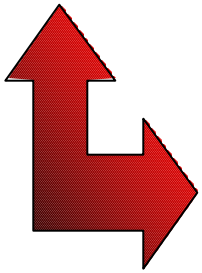
PARCEL TABLE

OBJECT_ID	PROJECTKEY	PARCELID	OWNERID	OTHER FIELDS
1	205 48 1	2371	1016	
2	205 131 8	2052	213	
3	237 23 3	2371	213	
4	262 203 1	2621	650	
5	909 185 2.03	9091	3074	
6	909 227 9	9092	3134	
6727				

OWNER TABLE

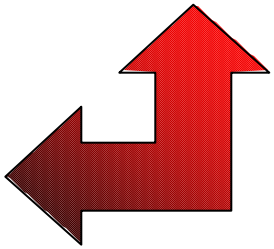
OBJECT_ID	OWNERID	COUNT#	ONWER NAME	ADDRESS
1	1016	38	BOROUGH OF CARLSTADT	
2	213	32	EMPIRE TRACK	
3	213	32	EMPIRE TRACK	
4	650	7	THE PORT AUTHORITY NY &NJ	
5	3074	2	NJMC	
6	3134	2	MORI PROPERTIES	
5666				

RELATIONSHIP 1:
PARCEL TO INTERMEDIATE TROUGH **PARCELID**



INTERMEDIATE TABLE

OBJECT_ID	PROJECTKEY	PARCELID	OWNERID
1	205 48 1	2371	1016
2	205 131 8	2052	213
3	237 23 3	2371	213
4	262 203 1	2621	650
5	909 185 2.03	9091	3074
6	909 227 9	9092	3134
6968			



RELATIONSHIP 2:
OWNER TO INTERMEDIATE TROUGH **OWNERID**